VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Fall Semester 2025-26

Lab Assignment – 2

Slot: L13+L14

Class: VL2025260105679

Branch: B.tech CSBS

Course code & title: CBS3005

Cloud, Microservices and Applications

Faculty name: Nithya K

DA by:

Kartikey Gupta

22BBS0105

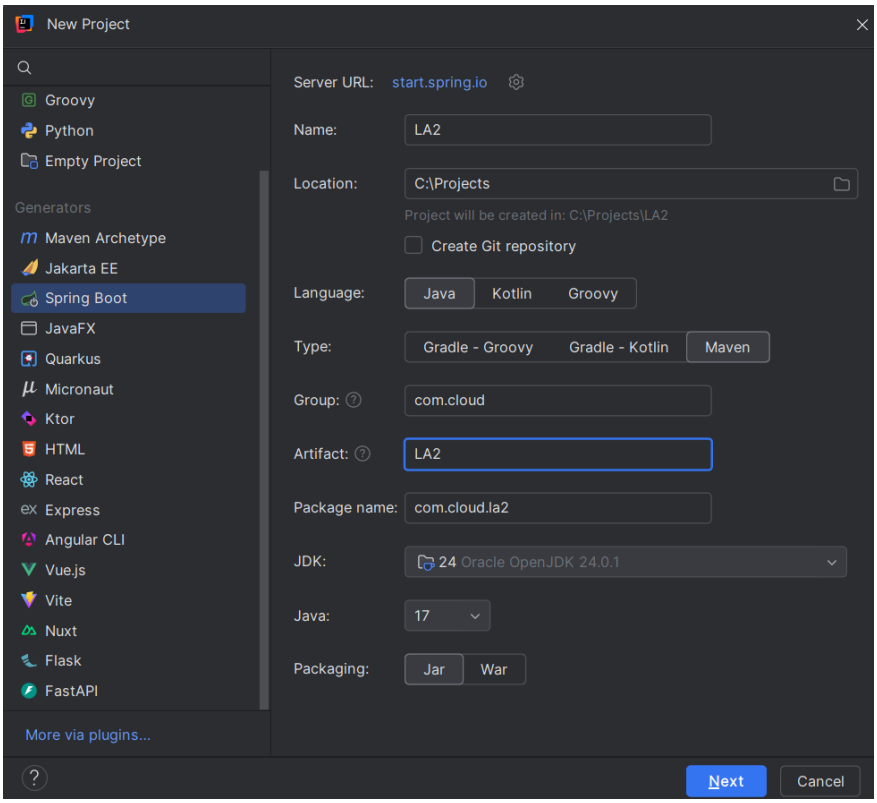
(i) Create a simple web application using your preferred programming language and framework (e.g., Node.js, Python, Java etc.). Ensure the application is fully functional and ready for deployment. Initialize an AWS Elastic Beanstalk environment for your application. Choose the appropriate platform (e.g., Node.js, Python, Java) and configure the environment settings. Package your web application and deploy it to the Elastic Beanstalk environment. Access the deployed web application via the Elastic Beanstalk URL provided. Test its functionality to confirm that the deployment was successful and that the application is accessible and performs as expected.

Step 1: Create a Simple Spring Boot Application

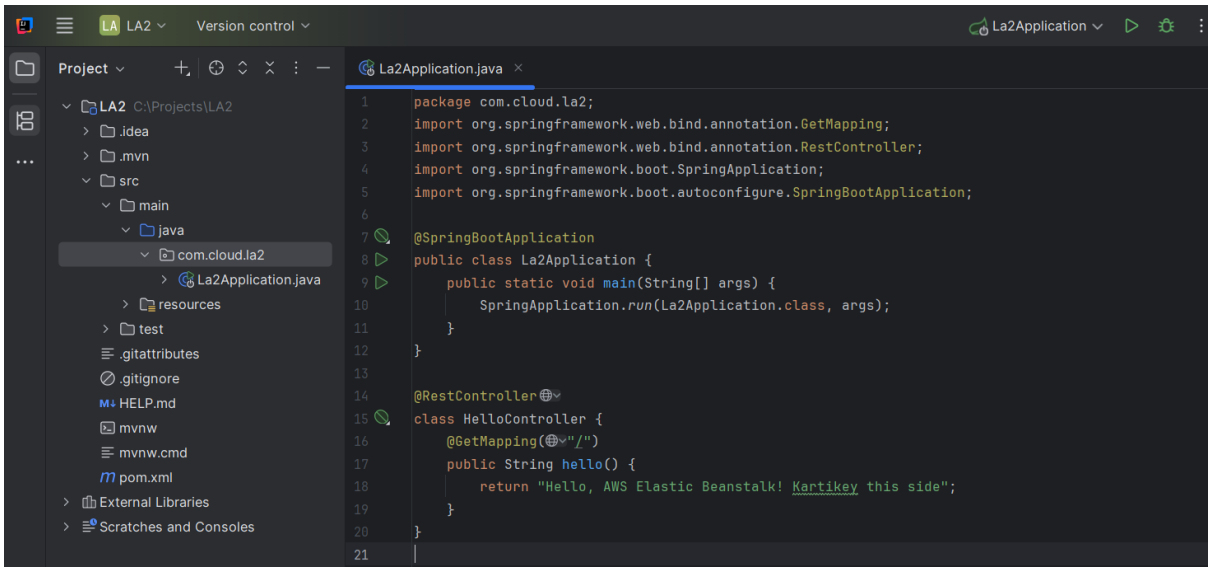
i. Open in your IDE IntelliJ, or VS Code.

ii. Create new springboot project
   - Project: **Maven**
   - Language: **Java**
   - Spring Boot: latest stable version
   - Dependencies: **Spring Web**

iii. Group: com.cloud, Artifact: LA2

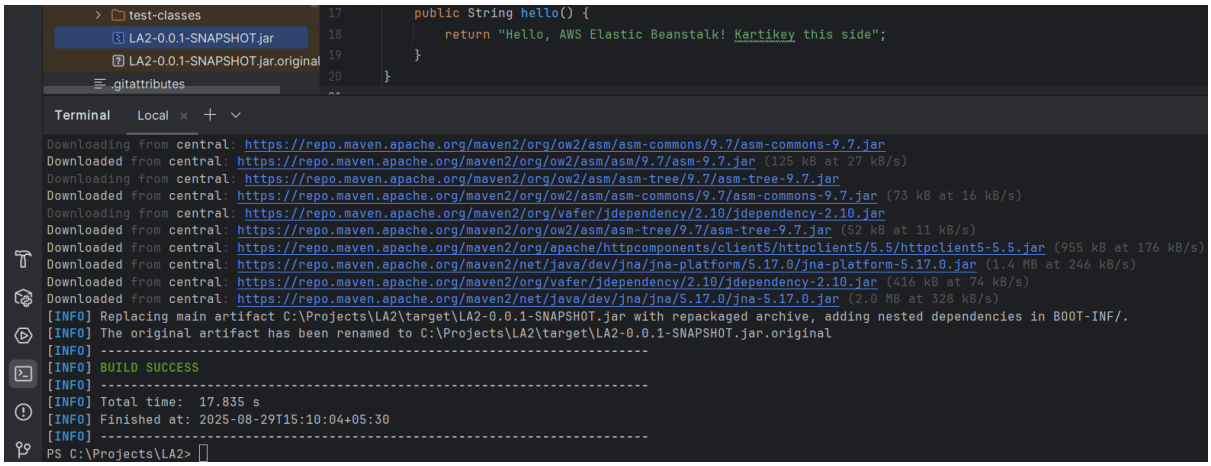


iv. Edit src/main/java/com/cloud/la2/La2Application.java to create a simple REST endpoint:



v. Clean the project using mvn clean package

vi. Build success and Jar file created



Step 2: Log in to AWS Management Console

i. Go to AWS Console.

ii. Log in with your AWS account.

iii. Navigate to **Elastic Beanstalk**

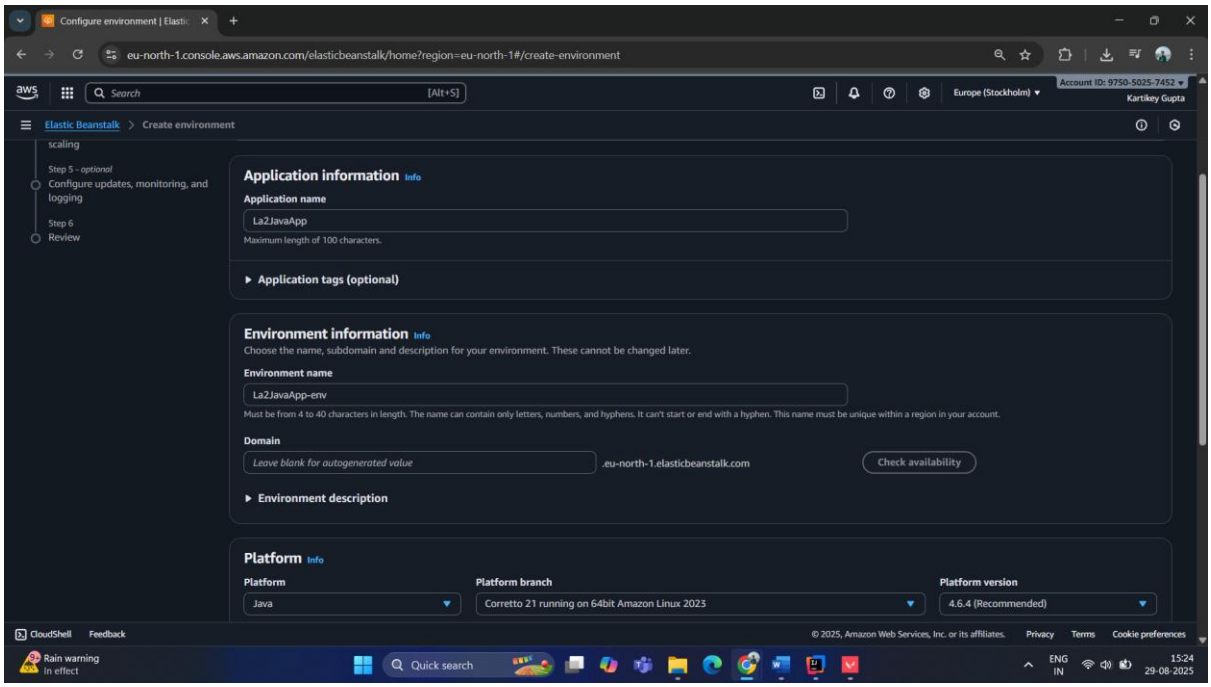
Step 3: Create a New Application

i. Click **Create application**.

ii. Enter **Application name**: La2JavaApp.

iii. Add an optional description: Sample Spring Boot app.

iv. Click **Create application**



Step 5: Configure Environment and Platform

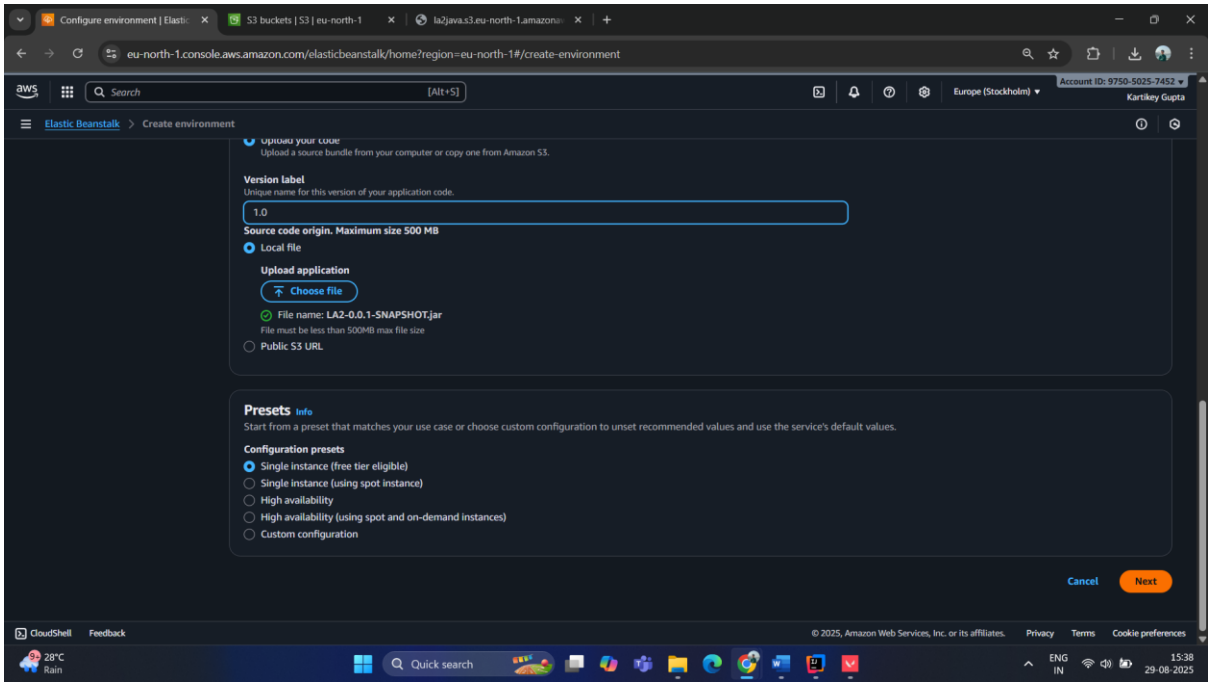
1. **Environment name**: La2JavaApp-env
2. **Platform**: Choose **Java**
    - For Spring Boot JAR: choose **Corretto 17** (or latest Java 17+)
3. **Platform branch**: Keep the default

Step 6: Upload Your Application

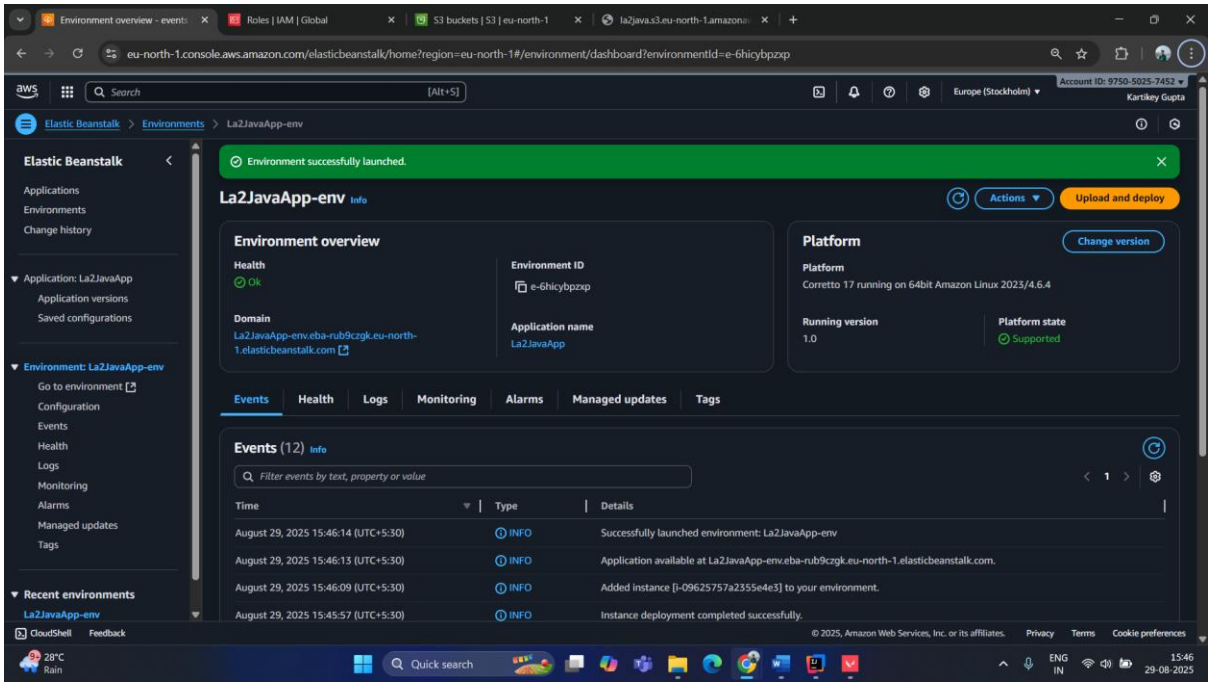
1. Under **Application code**, choose **Upload your code**.
2. Click **Choose file** and select your **JAR file** from target/.

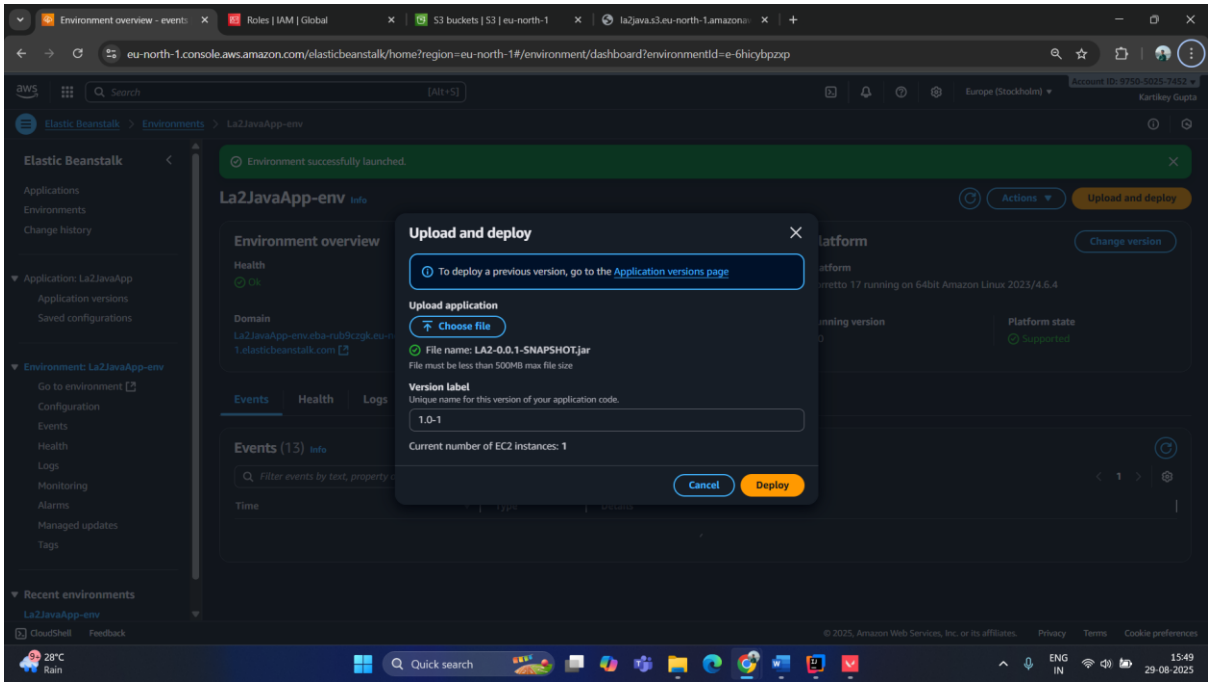
Step 7: Optional Environment Settings

1. **Instance type**: For testing, choose t2.micro (free tier eligible).
2. **Capacity**: Keep single instance.

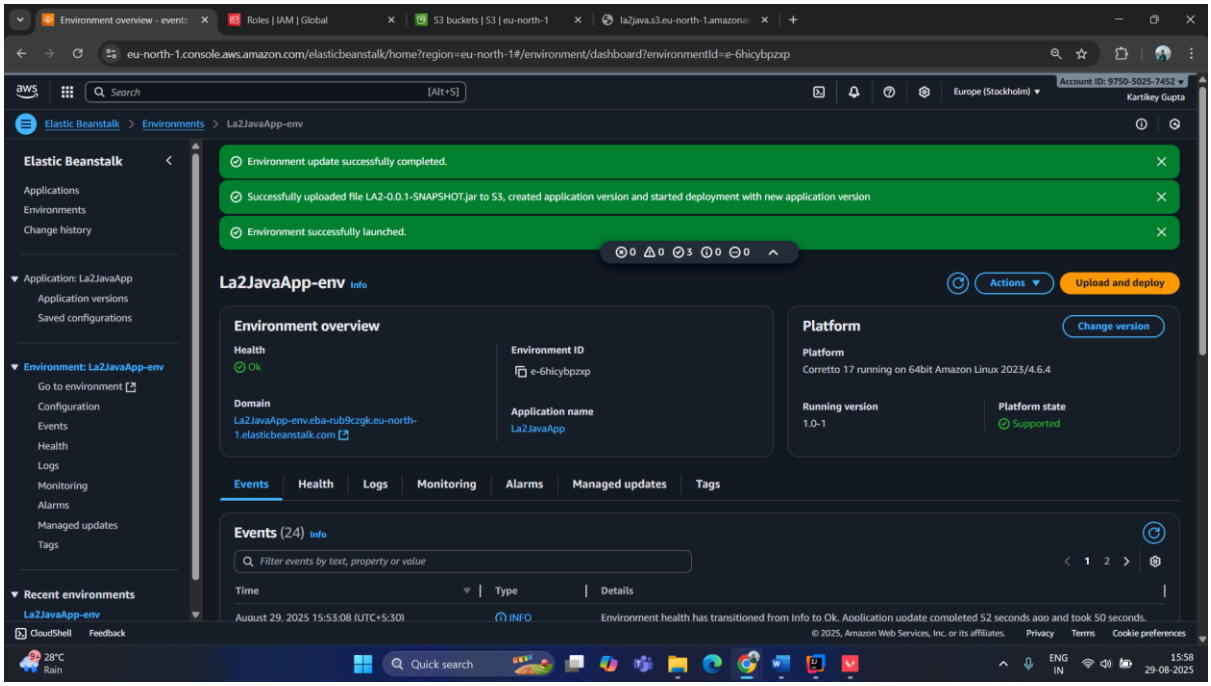


Uploading JAR file:



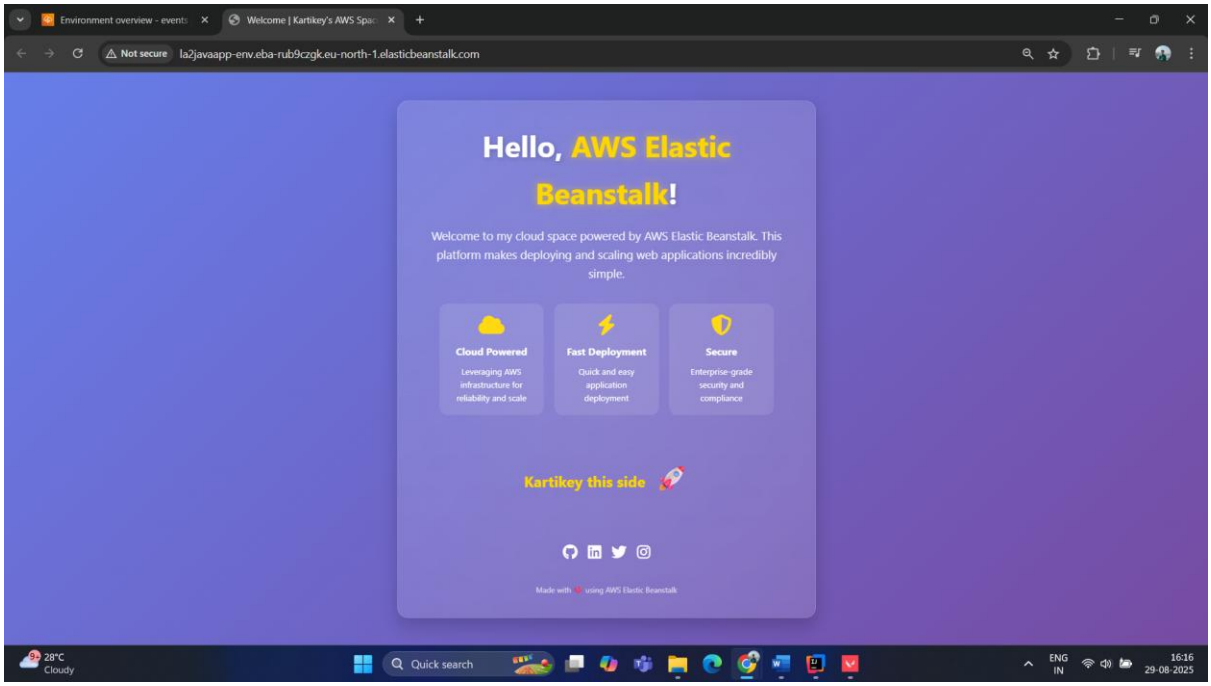


Environment created successfully



Accessing using the link: (will disable later due to incurring charges)

http://la2javaapp-env.eba-rub9czgk.eu-north-1.elasticbeanstalk.com/



Deployed web application successfully via the AWS Elastic Beanstalk!!